

Amendments to the Specification

This paragraph will replace all prior versions and listings of paragraph 0011 in the application:

FIG. 2 illustrates a flowchart of method steps in the preferred embodiment of the present invention. The validity flag initially is cleared or reset, signifying that there is no command to be processed in the queue. The queue of commands is filled partly or entirely. Each queue entry that has a valid command also has a validity flag set. The first command M is retrieved 115. If it is determined that there is a next valid write command 120, processing commences for the next write command 130 as indicated by a Next Valid Write Address pointer contained within the queue entry of command M. Data is accumulated through the data pipeline during this time. Write Look Ahead (WLA) occurs even during a write command. That is, if write B immediately follows write A, the pipeline will fill data for write B as soon as it has 1) fetched the final data for write A and 2) there is available space. Preferably, the write look ahead processing occurs in parallel to normal command processing and may be considered a part of pipeline logic. The outbound pipeline logic begins filling for a first write command A, and only stops when it reaches the end of that command. Filling will pause when the pipeline becomes full and will resume after the data has been written on the bus. Only when the command processing logic reaches write command A will the Next Valid Write Address pointer reflect any possible write command B – at which point the pipeline logic will again begin filling – this time for write B. (Note that while each entry in the command queue has its own Next Valid Write Address pointer, the command processing logic uses the Next Valid Write Address pointer associated with the current command.) In another embodiment, when command M is retrieved, any concurrent write operations are terminated. The operation is performed and the validity flag for that queue entry is cleared 140. If command M is a ~~write~~ read command, the ~~write~~ read operation is performed according to the op code and parameters stored in the queue entry for command M 135. If command M is not completed because of a time out, the detection of data corruption, or another fault condition 140, an optional REDO flag may be set 145. In an optional redo process, the commands that are determined to have failed may be

processed again in another pass through the queue. The present invention even allows multiple passes through the queue to allow processing of all stored commands. A limit may be placed on the number of redo stages for a queue to avoid system hang-ups. For example, a counter may be employed to count the number of stages per queue. If a threshold is reached, the processor issues an alert to an operator, enacts a bypass procedure, or performs other operations. A counter threshold may be hardwired or presettable by the processor, either automatically or through operator entry. If the last command in the queue has been traversed 150, other processing 160 may be performed. The other processing includes the optional redo process. Otherwise, the next queue entry is retrieved 155.